

1 **CLAIMS**

2
3 1. One or more processor-accessible media comprising processor-
4 executable instructions that, when executed, direct a device to perform actions
5 comprising:

6 accepting a plurality of files, each file of the plurality of files corresponding
7 to a respective file type and including source code;

8 associating a build provider with each file of the plurality of files in
9 accordance with the corresponding respective file type;

10 ascertaining the source code of each file of the plurality of files via the
11 associated build provider; and

12 compiling the ascertained source code of each file of the plurality of files
13 into an assembly.

14
15 2. The one or more processor-accessible media as recited in claim 1,
16 comprising the processor-executable instructions that, when executed, direct the
17 device to perform a further action comprising:

18 accessing a data structure that maps respective file types of a plurality of
19 file types to respective build providers of a plurality of build providers.
20
21
22
23
24
25

1 3. The one or more processor-accessible media as recited in claim 1,
2 comprising the processor-executable instructions that, when executed, direct the
3 device to perform a further action comprising:

4 instantiating the associated build provider for each file of the plurality of
5 files.

6
7 4. The one or more processor-accessible media as recited in claim 1,
8 wherein the action of compiling comprises an action of:

9 launching a compiler that performs a compilation on the ascertained
10 source code of each file of the plurality of files to create the assembly.

11
12 5. The one or more processor-accessible media as recited in claim 1,
13 wherein at least a portion of the processor-executable instructions comprise at
14 least part of an operating system.

15
16 6. The one or more processor-accessible media as recited in claim 1,
17 wherein at least a portion of the processor-executable instructions comprise at
18 least part of a program that is capable of establishing a runtime environment.

19
20 7. The one or more processor-accessible media as recited in claim 1,
21 wherein the one or more processor-accessible media comprise at least one of (i)
22 one or more storage media and (ii) one or more transmission media.

1 8. One or more processor-accessible media comprising processor-
2 executable instructions that, when executed, direct a device to perform actions
3 comprising:

4 creating an associated build provider for each associated file of a plurality
5 of files;

6 giving each associated build provider a path to its associated file;

7 requesting each associated build provider to contribute code of its
8 associated file; and

9 compiling the contributed code of each associated file into an assembly.
10

11 9. The one or more processor-accessible media as recited in claim 8,
12 comprising the processor-executable instructions that, when executed, direct the
13 device to perform a further action comprising:

14 accepting the plurality of files, each file of the plurality of files
15 corresponding to a different file type.
16

17 10. The one or more processor-accessible media as recited in claim 9,
18 wherein the action of creating further comprises an action of:

19 instantiating the associated build provider for each associated file of
20 the plurality of files according to the corresponding different file type of
21 each associated file.
22
23
24
25

1 **11.** The one or more processor-accessible media as recited in claim 8,
2 comprising the processor-executable instructions that, when executed, direct the
3 device to perform a further action comprising:

4 asking each associated build provider for its usable code language.
5

6 **12.** The one or more processor-accessible media as recited in claim 8,
7 comprising the processor-executable instructions that, when executed, direct the
8 device to perform a further action comprising:

9 receiving one or more resources from at least one associated build provider.
10

11 **13.** The one or more processor-accessible media as recited in claim 12,
12 wherein the action of compiling further comprises an action of:

13 compiling the contributed code of each associated file and the one or
14 more resources from at least one associated build provider into the
15 assembly.
16

17 **14.** The one or more processor-accessible media as recited in claim 8,
18 wherein the action of compiling further comprises an action of:

19 constructing at least one of an object code file, an executable file, a
20 dynamically linked library (DLL) file, and an intermediate language (IL)
21 file.
22
23
24
25

1 **15.** The one or more processor-accessible media as recited in claim 8,
2 wherein the action of giving further comprises an action of:

3 calling a file path interface on each associated build provider.
4

5 **16.** The one or more processor-accessible media as recited in claim 8,
6 wherein the action of requesting further comprises an action of:

7 calling a generate code interface on each associated build provider.
8

9 **17.** The one or more processor-accessible media as recited in claim 8,
10 comprising the processor-executable instructions that, when executed, direct the
11 device to perform a further action comprising:

12 acquiring the contributed code of each associated file via each associated
13 build provider responsive to the action of requesting.
14

15 **18.** The one or more processor-accessible media as recited in claim 17,
16 wherein the action of acquiring further comprises at least one of the following
17 actions:

18 retrieving the contributed code from a stipulated path location;

19 retrieving the contributed code from a created code object; and

20 retrieving the contributed code as a code compile unit.
21
22
23
24
25

1 **19.** A device comprising:
2 at least one processor; and
3 one or more media including a data structure that is capable of being
4 accessed by the at least one processor, the data structure comprising:

5 a first entry that includes a first file type and a denotation of a first
6 build provider, the first build provider adapted to handle files of the first
7 file type during a compilation;

8 a second entry that includes a second file type and a denotation of a
9 second build provider, the second build provider adapted to handle files of
10 the second file type during a compilation; and

11 a third entry that includes a third file type and a denotation of a third
12 build provider, the third build provider adapted to handle files of the third
13 file type during a compilation;

14 wherein the first entry maps the first file type to the first build
15 provider, the second entry maps the second file type to the second build
16 provider, and the third entry maps the third file type to the third build
17 provider.

18
19 **20.** The device as recited in claim 19, wherein the first build provider is
20 capable of generating source code from files of the first type, the second build
21 provider is capable of generating source code from files of the second type, and
22 the third build provider is capable of generating source code from files of the third
23 type.

1 **21.** One or more processor-accessible media comprising a build
2 provider that is tailored for a particular file type, the build provider adapted to
3 generate code from files corresponding to the particular file type and to contribute
4 the generated code to a compilation.

5
6 **22.** The one or more processor-accessible media as recited in claim 21,
7 wherein the build provider is capable of interacting with other software that is
8 capable of compiling an assembly involving multiple respective files that
9 correspond to multiple respective file types.

10
11 **23.** The one or more processor-accessible media as recited in claim 21,
12 wherein the build provider is configured to contribute code for compilations, the
13 compilations involving files that correspond to other files types.

14
15 **24.** The one or more processor-accessible media as recited in claim 23,
16 wherein the build provider is configured to contribute code for compilations (i) by
17 writing to a code file object, (ii) by writing to a stipulated file path location, and/or
18 (iii) by generating a code compile unit that presents code as a language-
19 independent structure.

1 **25.** One or more processor-accessible media comprising software that is
2 adapted to create, maintain, and/or use a data structure that has a plurality of
3 entries, each respective entry of the plurality of entries including a respective file
4 type and a denotation of a respective build provider; the respective build provider
5 adapted to generate code from files of the respective file type and to contribute the
6 generated code to a compilation.

7
8 **26.** The one or more processor-accessible media as recited in claim 25,
9 wherein the data structure comprises a build provider registration mapping data
10 structure.

11
12 **27.** The one or more processor-accessible media as recited in claim 25,
13 wherein the software is further adapted to orchestrate the compilation, the
14 compilation involving multiple files corresponding to multiple file types; each
15 respective file type of the multiple file types having a respective entry in the
16 plurality of entries.

1 **28.** One or more processor-accessible media comprising processor-
2 executable instructions that expose an application programming interface (API),
3 the application programming interface including:

4 a first property that is adapted to accept a path that identifies a file to which
5 the processor-executable instructions is to be associated;

6 a second method that is adapted to indicate a language used by code that is
7 part of the file; and

8 a third method that is adapted to precipitate (i) generation of the code from
9 the file and (ii) contribution of the generated code to a compilation.
10

11 **29.** One or more processor-accessible media comprising processor-
12 executable instructions that, when executed, direct a device to perform actions
13 comprising:

14 receiving a path that identifies a particular file having particular code and
15 corresponding to a particular file type, the processor-executable instructions
16 corresponding to the particular file type; and

17 contributing the particular code of the particular file to a compilation
18 involving multiple files.
19

20 **30.** The one or more processor-accessible media as recited in claim 29,
21 comprising the processor-executable instructions that, when executed, direct the
22 device to perform a further action comprising:

23 indicating a usable code language, the usable code language comprising the
24 language of the particular code.
25

1 **31.** The one or more processor-accessible media as recited in claim 29,
2 comprising the processor-executable instructions that, when executed, direct the
3 device to perform a further action comprising:

4 generating the particular code from the particular file.
5

6 **32.** The one or more processor-accessible media as recited in claim 29,
7 comprising the processor-executable instructions that, when executed, direct the
8 device to perform a further action comprising:

9 submitting one or more or resources for inclusion in the compilation.
10

11 **33.** The one or more processor-accessible media as recited in claim 29,
12 wherein the processor-executable instructions correspond to the particular file type
13 such that the processor-executable instructions are capable of generating source
14 code from files generally that correspond to the particular file type.
15

16 **34.** The one or more processor-accessible media as recited in claim 29,
17 wherein the action of contributing comprises at least one of the following actions:

18 writing the particular code to a created object;

19 writing the particular code to a stipulated path location; and

20 using a code object model provider to generate a code compile unit
21 from the particular code.
22
23
24
25

1 **35.** An arrangement for software build extensibility, comprising:

2 association means for associating a build provider with each respective file
3 of a plurality of files in accordance with a respective file type that corresponds to
4 the respective file;

5 ascertainment means for ascertaining code of each respective file of the
6 plurality of files via the associated build provider; and

7 compilation means for compiling the ascertained code of each respective
8 file of the plurality of files into an assembly.

9
10 **36.** The arrangement as recited in claim 35, further comprising:

11 contribution means for contributing the code of each respective file of the
12 plurality of files to the ascertainment means as at least one of a code object, a file
13 path location, and a code compile unit.

14
15 **37.** The arrangement as recited in claim 35, wherein the arrangement
16 comprises at least one device.

17
18 **38.** The arrangement as recited in claim 35, wherein the arrangement
19 comprises one or more processor-accessible media.
20
21
22
23
24
25